APRIL 1981

UQ CAMOL NEWSLETTER

EDITION 1. UNIVERSITY OF QUEENSLAND

IN THIS ISSUE

UQ CAMOL PROJECT COMMENCED

Dr Brian Carss said that a major aim of the project was to combine the attributes of micro computer systems and large central systems to provide a system tuned to student and teacher needs. He emphasised the need to involve University teaching staff during the development and implementation stages.

CAI FOR FIRST LEVEL COMPUTER SCIENCE

Adequate assistance to students, common standards of education and the problems of assessment for the 600 students in CS100 "Introduction to Computer Science" are some of the problems which Marshall Harris hopes to reduce with a CAI system.

THE APPLE II MICRO-COMPUTER

Geoff Vandenberg provides a review of the system. Cathy Chapman's function plotter is described. The availability of a Spanish Language Instruction package and a Group Velocity demonstration package is reported.

TOOLKIT AVAILABLE

"Toolkit" is the first operational part of the UQ CAMOL system. By simple menu operation from an Apple computer, programs can be selected from a library of instruction programs on the Central Computer and loaded on to an Apple. Try it out.

COMPUTERS HELP HANDICAPPED PEOPLE

John Barker describes a cooperative project by the Queensland Spastic Welfare League, the Prentice Computer Centre and the Stones Corner Rotary Club.

APPLE PILOT

The PILOT language is in widespread use for creating CAI material. Apple PILOT is described.

1.0 Edition I

1.1 Introduction

The CAMOL Newsletter is aimed at improving communications between people engaged in computer based learning within the University. It is a fast expanding area and there will be a need, if we are to be effective, to balance individual initiatives against the requirement to provide facilities at low cost across the total University.

The University and the Tertiary Education Committee has, through its Development Grants, provided a significant impetus to development. The grants approved in 1981 are as follows:

DEPARTMENT	SUBJECT	1981 APPROVED FUNDING
University	Appointment of C.A.L. Advisor	$15,000
Faculty of Education Prentice Computer Centre	Infrastucture for a Computer Assisted Management of Learning	51,180
Commerce	Managerial Accounting and Introductory Financial Accounting	22,402
Computer Science	First level Computer Science Courses	43,046
Chemistry	Modification of diagnostic tests and initial remedial tutorials for the CALCHEM system	5,750

These are not, of course, the only developments and we hope through this newsletter to publish information of what is going on across all departments of the University. This, of course, will be dependant upon contributions. Contributions will be published without editorial changes but we suggest that they be no longer than two pages of a newsletter. Of course, short statements of a couple of paragraphs would be welcome.

The editor of the CAMOL Newsletter is Geoff Vandenberg, Prentice Computer Centre (telephone extension 3021), and comments and contributions should be directed to Geoff.

Alan W. Coulter,
Director.

1.2 The CAMOL- UQ Project

The CAMOL-UQ acronym is translated into Computer Assisted Management of Learning at the Universtity of Queensland. This is a joint project of the Education Department and the Prentice Computer Centre funded through a General Development Fund grant.

The aim of this project is to develop the necessary infrastructure to enable the simplicity of operation of a micro-computer to be combined with the large resources and great flexibility that are available to the user of a large computer. The developmemt of an infrastructure will enable us to combine the attributes of a large computer and a micro-computer to minimise their individual limitations and make power available to the user in such a way, that the complex operating and communication protocols are completely transparent.

Progress to date is summarized below:

(a) A TOOLKIT data-base structure design and implementation has been completed, which enables users to search for and down load any file from the PDP-10 to an Apple II micro-computer.

(b) A function decoder, supporting a menu-type operation and plotting routines are also operational on the Apple II. The function decoder routine is based on the algorithm described by Calingaert (1979).

(c) The design characteristics of a TEACHER data-base to contain structured lesson materials have been developed.

The infrastructure that is being developed will consist of a data-base of instructional tools, lesson materials and student evaluation data. Users will have available, the file transfer facilities of the data communications network and be able to use the local intelligence of a micro-computer system to provide a terminal tuned too student and teacher needs.

One of the major reasons for the failure of the early attempts with computer assisted learning (CAL) was teacher rejection on the basis of disillusionment with the poorly implemented CAL packages. It is accepted that the best way to gain teacher acceptance of materials presented in this way is to have the maximum teacher involvemamt in the project. We propose to involve teaching staff through this newsletter, seminars, workshops, and seeking of assistance with system trials during the development stages of the project.

Brian W. Carss and Alan W. Coulter,
Education Department -- Prentice Computer Centre.

1.3 Apple II Micro-Computer, a Review

The Apple II is a complete system with extensive software available all contained in a neatly presented package. It is capable of taking up to 48K of memory with an additional 16K in a language card (of which only 6K is available to the user). The system monitor supports commands that examine and deposit data into memory, move and compare blocks of memory, load and store blocks on cassette, diskettes, assemble and dissemble op codes, run trace and single step programs, display and modify 6502 registers, and perform hexadecimal arithmetic.

Displays are fast and color is available. Display options include 24 lines by 40 uppercase characters or 40 by 40 low-resolution or high resolution graphics with 15 and 7 colors respectively. The high resolution graphics is 280 horizontal by 160 (or 192) vertical points.

There are eight periphal connectors for devices such as floppy disk controllers, RS232c serial interface cards (for connection other devices such as a printer or the PDP-10), special purpose language cards (Pascal-Fortran), 16K ram cards, color monitor display cards, 80 character upper-lower case expansion cards and even a card to de-Apple your Apple!, which makes you Apple look like a Z80 based CP/M Micro.

Of course the famous "Space Invaders" is available for the Apple but under the name "Steller Invaders". This game is good therapy for jaded lecturers and their kids!

We have found the the Apple is reasonably reliable and performs quite adequately for a micro based on the 6502 (a little dated by the Z80a, which runs at four times the speed). The five-inch floppy disks appear to be much the same as on other systems in terms of realiability but lack a little in efficient storage density (about 120,000 bytes per floppy).

A word of warning, the floppy disks MUST be handled correctly. Damaged floppy disks are at the bottom of nearly ALL disk problems I have been asked to look at. So, keep them out of the sun, rain, away from electrical devices, pizzas, milkshakes, kids, cigaratte ash, wine and beer soaked table tops etc.

Unfortunately most of the operating systems on the Apple are incompatible. Pascal disks cannot by used under DOS version 3.2 or DOS 3.3 or CP/M etc, etc. This is a little dissappointing as the packing densities used are nearly all identical.

The standard programming language on the Apple is Basic. Both Floating point and the so called Fast Integer Basic are available. The Basic is interperative and execution is fast and clean. A certain amount of syntax checking is performed when entering statements. Programs are saved on disk in a special tokenised form which saves space but prevents simple listing of the program (we have a program to take care of that problem).

Disk files are stored in various formats, some of them being "I" for tokenised Integer Basic, "A" for tokenised Applesoft (floating point basic), "T" for ascii text files. A full explanation is provided in the Apple manuals.

Documentation is excellent and well graded for easy starting with the Apple system. A very impressive range of "canned" software is available which includes word processing, accounting, teaching and games.

Pascal is a compiler version which compiles to P-Code, which is then interpreted. This results in usually slower execution for Pascal programs as compared with Basic (the problem being the Pascal compiler is not very efficient and the slowness of the interpreter). The same system is used for Fortran.

The ease of use of these three languages makes the Apple system an excellent tool for the tuition of these languages.

In summary a very neat tight package that is versatile and easy to use. My only criticisms of the Apple are the poor usage of disk space, and non-compatible disk formats. An excellent tool for the educator and student but not really up to standard for commercial applications.

G. A. Vandenberg,
Prentice Computer Centre,

1.4 CAI in the Computer Science Department

The Computer Science Department has, this year, for the first time , imposed a Semester 1 quota of 600 students for the first level course CS100, "Introduction to Computer Science". Even with this quota resources will be stretched beyond limits, and strains will be imposed upon academic staff. CS100 has a complement of one lecturer and several part-time, rather inexperienced, tutors. Being a practical subject, CS100 demands of its students 3 or 4 non-trivial programming assignments, which should be properly assessed and marked.

A cursory scan of each asssignment, without adequate tutorial comment, takes at least 5 minutes per student program. So far 600 students, each doing 4 assignment, the assessor must spend some 200 hundred hours marking. this can be shared, but this raises problems in maintaining a common standard among different markers/tutors.

In addition, staff must be available for consultation, and at present much advice and assistance has to be handled by inexperienced student tutors. Ideally , academic staff should carry out all the functions mentioned above, but the overload involved would effectively prevent staff from conducting any research. Indeed, even with

tutorial assistance from advanced students, staff workloads are excessive. Above all, and most seriously, students are not receiving the level of assistance and standard of education to which they are entitled.

It is this situation which has led the Computer Science Department to seek, successfully, funds to introduce a CAI (Computer Aided Instruction) system for 1st level Computer Science courses.

A full time Research Fellow (myself) has been appointed, to handle curriculum and teaching modules, and later on, a Systems Programmer will join the project to evaluate and adapt existing CAI systems, and/or to originate new software.

The Department currently has 90 micro-computers in use for editing, entering, and running student programs. We are exploring the possibilities of providing CAI via this existing set-up.

Initially, however, a limited experiment will be carried out using a small number of terminals, probably connected directly to a PDP-10 using the Gnosis CAI system, which was developed by Jacob Palme of the Swedish Research Institute of National Defence, and Dr Walter Manes of the Department of Philosophy, Old Dominion University, Norfolk, Virginia, USA.

GNOSIS is a system which provides reasonable facilities for authoring lessons, together with a fairly rudimentry ability to gather information on each student's performance. GNOSIS generates Algol-60 programs from the authoring language, which makes it easy for the teacher and systems programmer to understand and amend. Furthermore, GNOSIS is exeptionally well documented. Whether it will be suitable for use on a large scale remains to be seen.

Marshall Harris
Research Fellow,
Department of Computer Science.

1.5 A Function Plotting Program in PASCAL for the Apple II

The following article was submitted by Miss Cathy Chapmann who has since left the Prentice Computer Centre. The program was developed on the Apple Micro-computer, which had two floppy disks, language card and 48K of memory (plus 6K available from the language card). The operating system was the Apple Pascal system. The program plus research took about 1 week to complete. The program is available from the Centre at no charge.

A program which plots any given function Y of one variable X, has been developed for the Apple. It is written in Pascal, therefore a language card is required in order to run it.

The following restrictions apply:

(1) The functions LOG (base 10 logarithm) and LN (natural logarithm) will not be available until they become available on the Apple Pascal System.

(2) The upper and lower bounds of the X and Y axis are at present confined to 'short' integers.

(3) Since Apple Pascal has no power operation, it has been necessary to write an elementary one for the program, and this procedure accepts only positive integer indices.

(4) Functions which are undefined at any point in the given interval will cause the program to abort.

The available operations are the arithmetic operations (+,-,*,/,) and the following unary operations:- square root, sine (degrees), cosine (degrees), exponentiation, arctan (degrees), absolute value, unary minus, and the two logarithm operations mentioned above (when they become available).

After each function is drawn, the user is given the opportunity to superimpose other functions on the same axes.

The program uses a shift/reduce parsing algorithm, as described in Peter Calingeart's, Assemblers, Compilers and Program Translation, Computer Science Press, Potomac lud., 1979, section 6.2.3, Operator Precedence Parsing.

1.6 Computer Aid for Handicapped People

This is a project to use computers to aid a special class of handicapped persons to overcome problems of communication which severely limit their ability to lead ordinary lives, to cope with educational opportunities, and to compete for employment at other than unskilled and completely unchallenging levels.

The project is aimed at those with upper limb and speech impairments but who have normal or above average intelligence. Many persons with cerebral palsy or `spastics' fall into this category. There are however many causes of this disability.

Initial work in this project was carried out as a co-operative activity by staff from The Queensland Spastic Welfare League, the Prentice Computer Centre - University of Queensland, and the Stones Corner Rotary Club, which is providing funds for the project.

While the initial work is being carried out in conjunction with the Spastic Centre, the results will be made available to all similar institutions or to handicapped individuals.

The problem for these people is largely one of communication and the problem arises because they can only write and speak with difficulty and sometimes not at all. Some of these people have no use of their hands but can use a head piece (which is a stick attached by a head band to the forehead), while others may have limited use of one or both hands. If there is a speech impairment as well then the natural development of their intellectual facilities is severely inhibited.

In an effort to overcome these difficulties, some people use typewriters or more lately other keyboard devices such as computer terminals or word processors.

Typewriters are only partially successful because difficulties are experienced in inserting paper, making corrections and other necessary physical manipulations. Computer terminals and word processors are in some respects better but they are expensive and not very portable and hence accessible to relatively few people. In either case, conventional keyboards all have the common problem of requiring multiple key depressions to achieve certain actions. For example, a typewriter requires a shift key to be depressed simultaneously with another key to get upper case or special characters.

A further problem is that actual entry of text or data is often laboriously slow for these people and this can tie up expensive equipment uneconomically.

The solution offered by this project is to provide handicapped people with inexpensive microcomputers containing software which has been appropriately modified.

After modification, all operations of the keyboard require only one key depression at any one time. The user can create text and edit or correct it on the screen and store it on an ordinary audio cassette. This can be done at home or anywhere else at leisure under no pressure. Subsequently, the cassette may be taken to a conventional word processing machine if required for further tidying up of the text, or the text may be printed as is on a printer. In this way a few centrally located relatively expensive word processing machines can service the needs of a relatively large number of handicapped persons.

To get the project going Stones Corner Rotary purchased an Exidy Sorcerer micro computer for prototype development. This was presented to John Earles, Deputy Director of the Spastic Centre, during 1980.

The standard software has been extensively modified and includes the sequential or single keystroke facility for all shift functions.

The Spastic Welfare League have several Vydec word processing minicomputers and the necessary work has been done in the SORCERER's to enable the transfer of text files back and forth between SORCERER's and VYDEC's.

Another software modification allows the keyboard to generate more than one character for each single keystroke. Thus often repeated words or phrases can be added to the on going text by depressing a special shift key followed by one other normal key. This facility is particularly useful to computer programmers. For example, the word "PRINT" or the phrase "PROCEDURE DIVISION" has to be typed often in the course of writing computer programmes using the Cobol language. The keyboard software in the microprocessor can be modified so that these words or phrases can be automatically generated by a single keystroke.

The object of this enhancement is to provide assistance to handicapped persons who have an interest in becoming computer programmers. This is a profession requiring mainly intellectual ability which many handicapped persons have, but which also requires ability to use a computer terminal or microcomputer keyboard with moderate efficiency. Software which reduces the number of keystrokes for a given output is thus potentially very useful.

Current effort in the project includes work on the recording method on the cassettes to make them more reliable, development of a better text editor, and a facility to allow users to set up their own `customized' tables of often repeated phrases to be generated by single keystrokes.

The progress with the prototype Sorcerer, encouraged the Spastic Centre to purchase five machines of their own. These machines have been modified according to the prototype design and they are currently in use at the Spastic Centre, Newfarm.

Thus a tool or facility is being developed of potentially great advantage to handicapped people. It would be of use internationally and at a price fairly soon that people are currently paying for colour television. In time one could progress to a situation where a very wide range of options would be open to a handicapped person for work or study at home, or any other place.

Another interesting aspect is that while automation based on computer technology is rapidly reducing job oportunities for some groups in the community, this project is enhancing the oportunities for an otherwise disadvantaged group of people.

Funds were initially required to purchase prototype equipment and in case any specialized software development was required which could not be obtained by voluntary effort. So far the purchase of the equipment has been the only cost. The plan for the remaining funds ($6000) is to set up a trust which would in turn provide funds for the following purposes:

- further development,
- promulgation of information about the project to other institutions and individuals,
- production and supply of the modified programs in the form of

memory `chips' for substitution in standard Sorcerers,

- promulgation of information about the project to other Rotary clubs, in Ausralia, and overseas.

Publicity for the project will initially come from press releases following formal handover of modified microcomputers and all distributed software will carry a Rotary banner. For example, all users of a microcomputer using the software will see the Rotary acknowledgement each time the computer is turned on.

J. W. Barker,
Prentice Computer Centre.

1.7 Computer-Assisted Instruction in Spanish on the Apple II

Robert Phillips, Miama (Ohio) University.

The need for efficient drill in learning a foreign language is an obvious case for utilizing the computer. Materials in this package for the Apple II provide (1) verb drills covering virtually all tenses of Spanish verbs (excluding the perfect subjunctive, and the future and conditional perfects), and (2) vocabulary drills which can be keyed to any textbook of the instuctor's choice.

The drills are written for the Apple II and require Applesoft Basic in ROM, DOS and 48K memory. The package from Conduit includes the programs on diskette (51/4") and an Instructor' Guide.

Available from: Conduit, P.O. Box 338, Iowa City, Iowa 52244 U.S.

1.8 Group Velocity Demonstration Using an Apple II

Of the many microcomputer-based units CONDUIT has reviewed, this demonstration of wave phenomena stands out from all others in its effective use of the medium. The program generates striking animated displays of wave motion that capture the user's interest immediately, and provoke a series of "What if?" questions that lead naturally into the exercises presented in the Student Guide.

In this package students study travelling waves of the form

$y=\cos(wt - kx)$

where y is the vertical amplitude, w is the angular frequency, t is the time, k is the wave number (the wave length), and x is the horizontal position. The high-resolution display of the Apple II allows this travelling wave to be easily displayed. The two game

paddles are used to vary the frequency and the wave number over a reasonable range of positive and negative values (-16 to +16). The student is reminded how to derive the expression v=w/k for the velocity of the wave, which can be verified experimentally by measuring it directly from the display screen.

The Student Guide describes in detail how students can use the program to illustrate wave group phenomena and experimentally verify basic relationships. The guide concludes with a discussion of several natural wave phenomena exhibiting group velocity effects, and presents a number of related exercises for the student.

The package from CONDUIT includes one copy of the Student Guide and the software on diskette for the Apple II (Applesoft in ROM, DOS, 48K).

The Computer Centre has placed an order for this package.

1.9 Apple Pilot, a CAI package for the Apple

The Apple becomes an even more effective teaching tool with the introduction of Apple Pilot. A high-level, easy to use language, Pilot was designed for use by educators, trainers, and courseware developers, and has been in use since 1968 as a language for creating Computer Assisted Instruction (CAI). In fact, Pilot is the most widespread CAI language available today.

Although the Pilot language does require training and practice, producing CAI materials with Pilot is claimed to be a simple task compared to other high-level languages. Courseware can be written which fully utilizes the capabilities of the Apple for sound and graphics.

Apple Pilot provides three extensions to common Pilot. These allow the creation of graphics, music and special characters without special programming. By using the graphics editor, the user can position points, draw lines and circles, fill in areas, choose colors and see immediate results. The final graphic is then named and referred to by name in the Pilot program.

Sound is another major extension that Apple Pilot offers. Music is composed or transcribed by specifying note selection, order, duration, and special effects, like warbling.

The third extension is the character set editor, which allows the user to draw and save special chatacters for specific programs. A Greek or Russian alphabet, for example, can be created and used for a language drill and practice. This feature was formerly offered on the only the most expensive CAI systems.

Apple Pilot system requirements are an Apple II, 48K memory, one disk drive, or two disk drives for Author mode and DOS 3.3.

Apple Pilot is now in use in several departments on the Campus and we hope to print a users review soon.

G. A. Vandenberg,
Prentice Computer Centre.

1.10 Toolkit System

Toolkit is the name for a data-base plus service routines for making available to Apple II micro-computers a suite of useful programs written in Applesoft Basic. The data-base resides on the PDP-10 with some of the service routines. The Apple requirements are: 1 disk drive, RS232c serial interface device, a line to the PDP-10, a PPN on the PDP-10 (KL), plus a copy of the special routines for loading the Apple on a floppy disk. Copies of this disk are available from the centre at no charge (bring you own disk). Currently the system only supports 16 sector disks, Applesoft with DOS 3.3.

The user connects the Apple II to the PDP-10, inserts the special Toolkit disk, and then boots the disk up. The communications software is loaded at boot time and is concealed from the user with the lowering of Himem. Once the boot is complete the user types the "&" command to the Apple prompt "]" causing the Apple to become a "dumb terminal". The user then logs on to the PDP-10 in the usual manner (no special PPN required). On completion of the PDP-10 login, by typing "V" (control V) the user returns to normal Applesoft on the Apple (the prompt "]" should be displayed). The user now runs the TOOLKIT menu from Apple disk, with the command "RUN TOOLKIT <CR>"

TOOLKIT will respond with a "#" prompt. Then the following options are available:

(1)	H	Help message.
(2)	Q	Quit from TOOLKIT Menu.
(3)	F	Find <named keyword>.
(4)	L	Load Current Toolkit being examined.
(5)	D	Display Toolkits located by a Find.
(6)	Z	Zero Pool of Toolkits to start new Find.
(7)	J	Jump to the next toolkit in the pool of finds.

The following is a typical example of using the TOOLKIT menu (Login to the PDP-10 assumed).

```
]RUN TOOLKIT

CAMOL TOOLKIT SYSTEM

#F PLOTX
```

```
TOOLKITS FOUND = 1

#D

DESCRIPTION: PLOTX IS A GENERAL PURPOSE FUNCTION PLOTTER
             FUNCTIONS MAY BE PLOTTED, SAVED ON DISK
             AND RESTORED OVER THE TOP OF OTHERS.
PRESS RETURN FOR INSTRUCTIONS : L (L causes a load to Apple)

LOADING

(PLOTX is now loaded direct into Apple memory)
```

While the program is being loaded the curser will appear to "be trying to dissappear"!, this is a good sign and means the load is working. If the line speed is slow (300 baud) the the load will take longer than the 25 seconds it takes to load PLOTX at 1200 baud. If the load seems not to be working (eg the curser is blinking in the normal manner or it has taken too long), then press RESET, and type "@" to return to dumb terminal mode and see if you are still connected by typing "T". If you have been disconnected, Login to the PDP-10 or attach to your PDP-10 job and start again from "RUN TOOLKIT" to Applesoft prompt "]". If you are persistently disconnected then contact me at the Centre (3021).

After the load is complete (signified by the backslash "\", then the usual Applesoft promt "]" the program is ready to be "RUN" or saved on disk just as if you had typed it in. It is advisable to save the program on your floppy if you intend to use it more than once. Do not forget to log off the PDP-10 after you have finished with the Toolkit date-base. Using the Apple as a dumb terminal will not effect most programs currently in Apple memory. Note, Integer Basic may destroy the communications software as will the commands "FP" or "INT" to the "]" or ">" prompts (the ">" prompt signifies Interger Basic is running in the Apple as apposed to "]" for Applesoft.

To return to the TOOLKIT menu simply type "RUN TOOLKIT". Some of the Toolkit entries automatically run the TOOLKIT menu on completion.

If you have a usefull Applesoft program for the Toolkit data-base you are most welcome to submit it for inclusion. If you have improved existing toolkits then these are also welcome. Forms for submitting a toolkit are available from me at the Centre.

G. A. Vandenberg,
Prentice Computer Centre. (3021)

Readers wishing to receive the CAMOL Newsletter regularly please complete the information below and return to Geoff Vandenberg, Prentice Computer Centre, University of Queensland.

CAMOL NEWSLETTER MAILING LIST

TITLE: ..... SURNAME: ..... INITIALS: .....

ADDRESS: .....

.....

NO OF COPIES: .....